DOT MATRIX PRINTER

OPTION

user manual

APPLE II INTELLIGENT PARALLEL INTERFACE KIT

EPSON

P8090027-4
Y490991010

EPSON

EPSON CORPORATION

EPSON OVERSEAS MARKETING LOCATIONS

EPSON AMERICA, INC. (L.A.)

3415 Kashiwa Street.
Torrance, CA 90505
U.S.A.
Phone: (213) 539-9140
Telex: 182412

EPSON ELECTRONICS
(SINGAPORE) PTE. LTD.
#09-13/#09-14, World Trade Centre,
1 Maritime Square, Singapore 0409
Phone: 2786071/2
Telex: RS 39536

EPSON DEUTSCHLAND GmbH
Am Seestem 24
4000 Düsseldorf 11,
F.R. Germany
Phone: 0211-5961001
Telex: 8584786

EPSON U.K. LTD.
Dorland House
1 st Floor, 388 High Road
Wembley, Middlesex, U.K.
Phone: (01) 900-0466/7/8/9
Telex: 8814169

PRINTED IN JAPAN

83.03-06

Table of Contents

INTRODUCTION

In this booklet the basic usage and the hardware of the interface kit APPLE II Intelligent Parallel Interface Kit (Interface Board: #8132, Interface Cable: #8231) specially designed for APPLE II are described. You are requested to read this booklet very carefully before using this interface to link an EPSON Dot Matrix Printer such as MX series Type III (GRAFT RAX PLUS in U.S.A.), FX series, RX series, etc. with APPLE II*. Misuse of this interface may damage the interface circuit or APPLE II. The interface kit (#8132 and #8231) permits a printer which has Bit Image Printing capability to produce hard copies of APPLE's High-Rsolution display partially or totally. It also enables EPSON Dot Matrix printers to perform text formatting operation.

*APPLE II is the registered trade mark of Apple Computr Inc.

EPSON CORPORATION reserves the right to make changes in the specifications in this manual at any time without notice.
In order to make improvement software in the P-Rom is also subject to change without notice. EPSON CORPORATION makes no warranties, either express or implied, in respect of this user's manual or in respect of the software described in this user's manual, its quality, performance, or fitness for any particular purpose. EPSON CORPORATION software is sold "As is". The whole risk as to its quality and performance is with the buyer. Some countries or states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you.

CONFIGURATION

Fig. 1 shows the APPLE II Intelligent Parallel Interface Kit which consist of the interface board (#8132) and the interface cable (#8231).

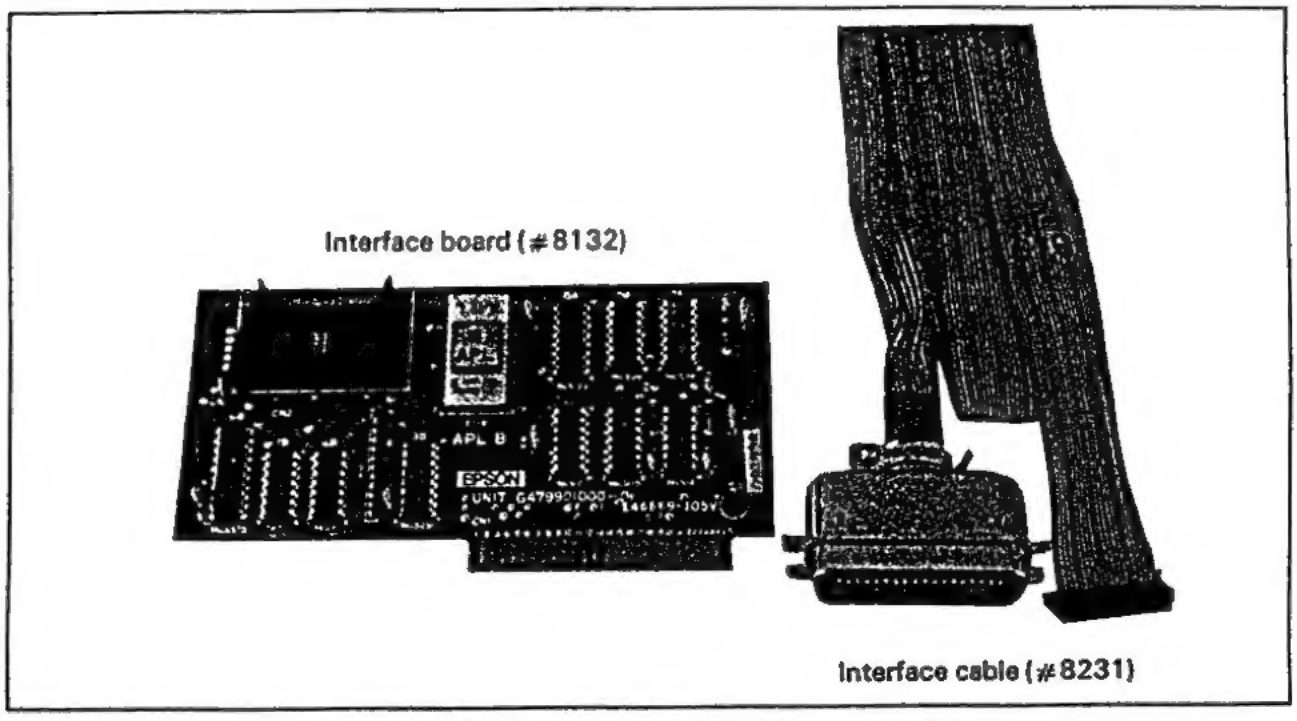


Fig. 1 configuration of the APPLE II Interface Kit Type 2

FUNCTIONAL SPECIFICATIONS

1. Text Printing

- Specifying the number of printing columns (40 to 132).
- Printing lowercase letters

2. Bit Image Printing

This interface can produce the following hard copies of the APPLE II CRT screen with EPSON Dot Matrix printers which has the Bit Image printing capability.

- Hard copy of the HI-RES display
  - 2.1. Normal printing
  - 2.2. Printing of a logical operation between HI-RES Page 1 and Page 2. (AND, OR, EOR)
  - 2.3. Inverse printing
  - 2.4. Printing by line unit
  - 2.5. Enlarged printing
  - 2.6. Dual density printing

3. Print Formatting

APPLE II BASIC does not offer a formatting function but the APPLE II Intelligent Parallel Interface Kit offers it as software.

- Integer representation
- Fixed point representation
- Exponent representation

INSTALLATION

Connect the interface cable to the interface board.

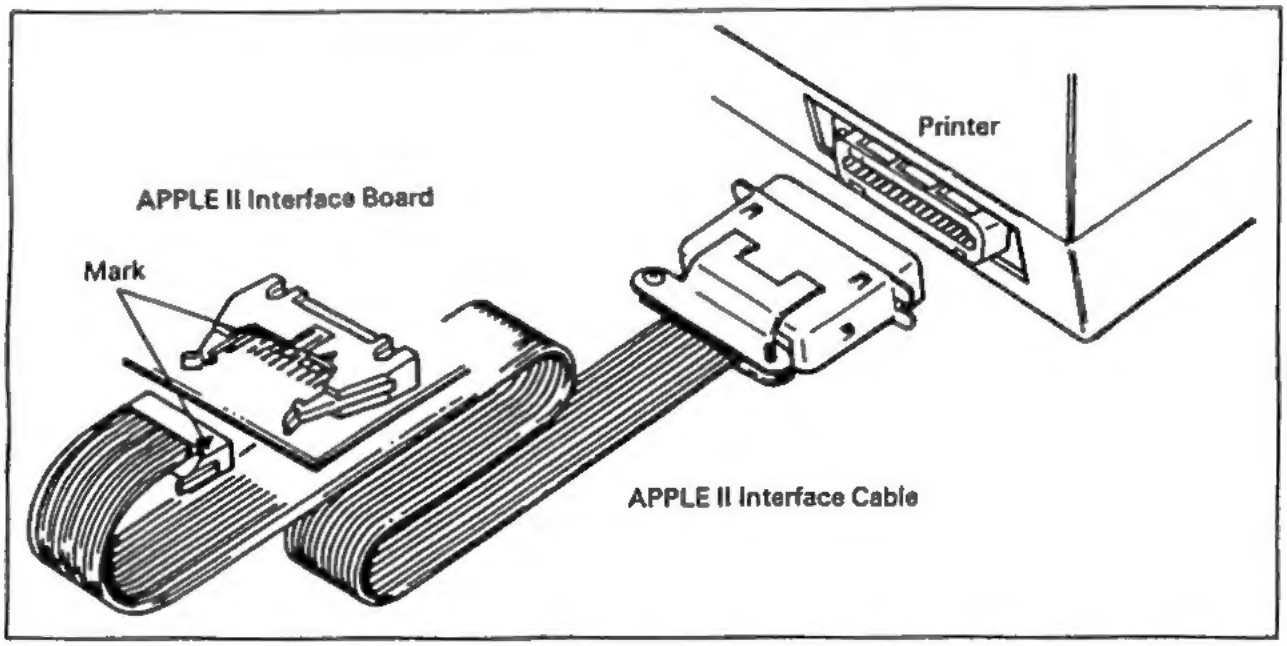


Fig. 2 Connection of the Interface Cable

Cut off the power supply to APPLE II and insert the interface board in a slot. If you forget to cut off the power supply, it may bring a serious damage not only to the interface board but also to the CPU and all ciruit boards in the system bus line. Make sure that no power is applied to APPLE II before connecting the interface board.
When the interface board is used in BASIC you may use any slot available for this connection. Install it to the slot "1" for the use of PASCAL or Z-80 Soft Card. In this manual all program samples and explanation are written assuming that the interface board has been placed in the slot "1".

Note: Be sure to turn on the power switch of APPLE II prior to that of the printer.

TEXT PRINTING

In this chapter, a description of the use of the interface is given. If you find the explanation difficult to understand, please familialize yourself with the APPLE II computer first.

1. Turning the Interface Board ON and Off

a. From BASIC mode
To use the interface board, it must first be turned on. The following is a description of the procedure commencing from either BASIC or Monitor mode.
Turn off: PR#0 (Turn off only the printer routine.)
POKE 53247, 0 (Also turn off formatting routine.)

b. From MONITOR mode
Turn on: nP^{C} (P^{C} denotes the Control P. Turn the interface board on from the Monitor)
Turn off: 0P^{C} (Turn off only the printer routine.)
Access of CFFF : (all routines go "off")
Note: This function will not be effective when DOS is in use.

2. Listing from BASIC

a. The number of columns (characters) per line is usually 40 on the TV monitor display and all program list are echo-backed on to it.

b. It is possible to specify up to 132 columns, but if the text exceeds 40 columns, it is not displayed on the CRT screen. When specification of more than 40 columns is desired, it should be written in decimal notation to the address adding "1656" (decimal notation) and "slot No."

Example:

```
10 SLOT = 1
20 REM LIST-MODE FOR 72 COLUMNS
        (Specifying 72 columns)
30 POKE 1656 + SLOT, 72
```

Note: The above-mentioned instruction (b) must be also carried out when the column displayed on the APPLE II CRT screen is different from the output to the printer. Turn on the interface board before specifying the number of columns by inputting PR#1.

3. Printing Lowercase Letters

Those letters which are enclosed by the preset control characters are printed in lowercase letters. The control character is normally set by W^c (CTRL W).

[Example]

```
]LIST

10  REM  SMALL LETTERS
20 :
30  PR# 1
40  PRINT "S"; CHR$ (23);"MALL LE
     TTERS "; CHR$ (23);"PRINTING
     "
50  PR# 0

]RUN

Small letters PRINTING
```

Note: In the DISK version, line 30 which turns the printer on should be as follows:
30 PRINT CHR$ (4); "PR#1"; CHR$ (13);
Also line 50 which turns the printer off should be
50 PRINT CHR$ (4); "PR#0".

4. Change of Lowercase Control Characters

When you want to change the control characters such as T^c and W^c to the others, you can do it in the following way;

[Example]
For lowercase characters;
POKE 1400 + SLOT, YY
(YY denotes a decimal notation of a desired code)

5. Double Printing

The APPLE II normally outputs an LF code immediately after it has output a CR code. Double printing may be performed when the LF code is inhibited. Write "255" to inhibit the LF code or "0" to release it in the address 1528 + SLOT No.

[Example]

```
]LIST

100  REM  DOUBLE PRINTING
110 :
120  PR# 1
130 SLOT = 1
140  POKE 1656 + SLOT,72: REM  OU
     T PUT PRINTER ONLY
150  POKE 1528 + SLOT,255: REM  K
     ILL LF
160  PRINT "NORMAL LETTERS "
170  PRINT  CHR$ (14);"          &
     "
180  PRINT  CHR$ (15); TAB( 35);"
      CONDENSED LETTERS"
185  PRINT  CHR$ (18)
190  POKE 1528 + SLOT,40: REM  SE
     ND LF CODE
200  POKE 11657 + SLOT,40: REM  O
     UTPUT SCREEN & PRINTER
210  PR# 0

]RUN


NORMAL LETTERS      & CONDENSED LETTERS
```

Note: It is possible to print in either enlarged and condensed on the same line by double printing technique.

6. Printing in Enlarged or Condensed Characters

this is achieved by applying the corresponding ASCII codes to the printer.

Enlarge Printing:	Setting	SO	CHR$ (14)
	Resetting	US	CHR$ (31)
Condense Printing:	Setting	SI	CHR$ (15)
	Resetting	DC2	CHR$ (18)

Refer to the program example shown in section 5.

7. Change of Line Spacing (or amount of paper feed)

The normal paper feed or line spacing is 1/6 inch or 1/8 inch. When necessary, you can change it dot by dot (by 1/72 inch) by providing a command signal lead by an ESCAPE code (hereinafter called "ESC").

a) ESC 0..... 1/8 inch
   CHR$ (27) + CHR$ (48)
b) ESC 2..... 1/6 inch
   CHR$ (27) + CHR$ (50)
c) ESC A + m ..... m/72 inch
   (Where 1 ≦ m ≦ 85)
   CHR$ (27) + CHR$ (65) + CHR$ (m)

The following is an example of variable line spacing which varies line spacing from 1/72" to 12/72" or 1/6" line by line.

[EXAMPLE]

```
]LIST

100  REM  VARIABLE LINE SPACING
110 :
120 SLOT = 1
130  PR# 1
140  POKE 1656 + SLOT,79: REM  K1
     LL CRT
150  FOR I = 1 TO 12
160  PRINT  CHR$ (27); CHR$ (65);
      CHR$ (I);
170  PRINT "VARIABLE LINE SPACING
     "
180  NEXT
190  POKE 1656 + SLOT,40: REM  WI
     TH CRT
200  PR# 0

]RUN


VARIABLE LINE SPACING
VARIABLE LINE SPACING
VARIABLE LINE SPACING
VARIABLE LINE SPACING
VARIABLE LINE SPACING
VARIABLE LINE SPACING
VARIABLE LINE SPACING
VARIABLE LINE SRACING
VARIABLE LINE SPACING
```

Note: When you change the line spacing in the above-mentioned way, you must set the printer for list print mode (POKE 1656 + SLOT, 72, etc.)

BIT IMAGE PRINTING

1. Bit Image Printing from BASIC

EPSON Dot Matrix Printer has the unique Bit Image printing function, which enables the print out versatile graphics.
Each needle of the print head can be selectively controlled by the data issued from the host computer (APPLE).

The bit image data consists of up to 8 bits of parallel data and their relation to each needle is shown Fig. 3.

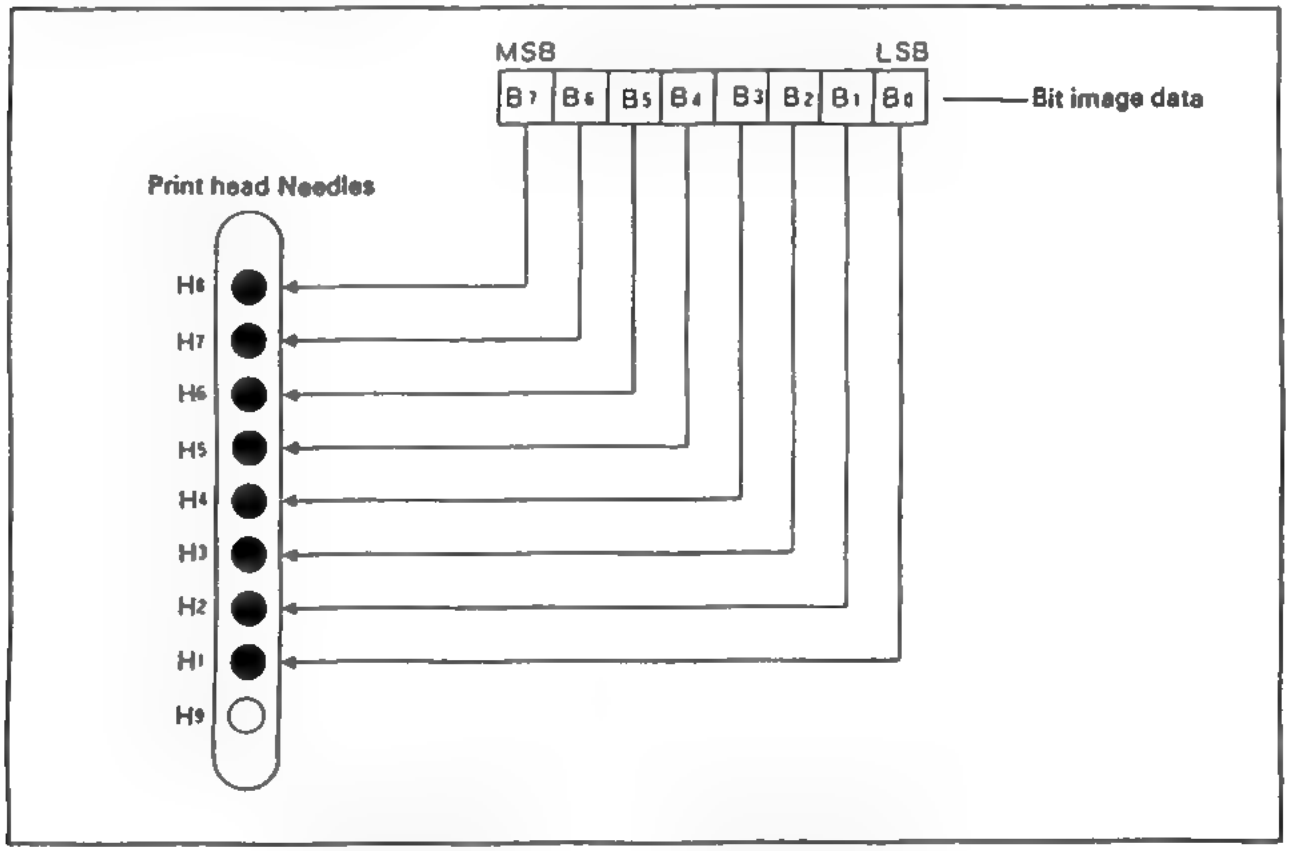


Fig. 3 Relationship between Needles and Bit Image Data

When your EPSON Printer is in the bit image mode, printing is performed from left to right in uni direction.
To let the printer go in this mode, it is necessary to specify the amount of bit image data after imput of ESC K code. The amount of data (number of bytes) to be transferred can be specified by first transfering the lower order 8 bits and then the upper order 8 bits.

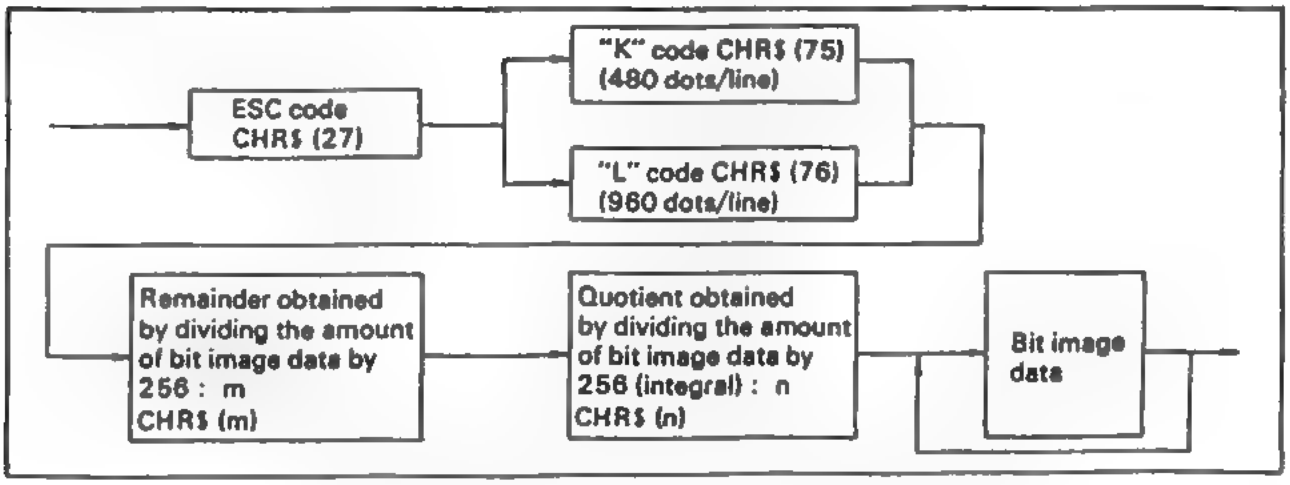
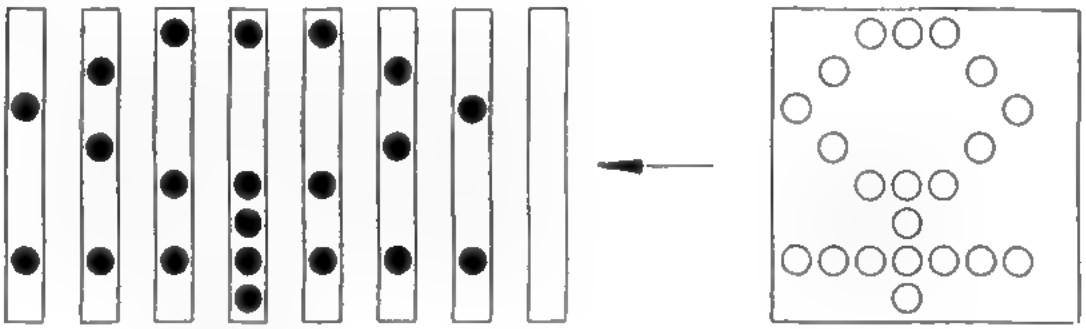


Fig. 4 Transfer Sequence of Bit Image Data

[Example]



$22, $52, $8A, $8F, $8A, $52, $22, $00 (Hexadecimal)

```
]LIST

100  REM  BIT-IMAGE PRINTING
110 :
120 SLOT = 1
130  PR# 1
140  POKE 1656 + SLOT,79: REM  KI
     LL CRT
150  PRINT  CHR$ (27); CHR$ (75);
      CHR$ (8); CHR$ (0);
160  PRINT  CHR$ (34); CHR$ (82);
      CHR$ (138); CHR$ (143); CHR$
     (138); CHR$ (82); CHR$ (34);
      CHR$ (0);
170  PRINT  CHR$ (10)
180  PR# 0
190  POKE 1656 + SLOT,40: REM  WI
     TH CRT


]RUN
```

The above program cannot print MSB data as it is masked.

Note: In this case, each code corresponding to the control characters such as Q^c, T^c, W^c, etc. may not be used. Refer to the section 2 for alternative methods. The printer will automatically return to the text printing mode after the specified number of data has been transferred.
APPLE BASIC cannot send 8 bit binary data, since the MSB of data is always "LOW". Therefore even if you make a program in APPLE BASIC, you can only specify the quantity of the Bit Image data less than 127 bytes.

2. Machine Language Routine to Transfer Data to Printer

Due to the characteristics of APPLE BASIC some of the data may not be transferred by BASIC statement CHR$ ( ). This insufficiency is avoided and all necessary data ($00 — $ FF) is transferred by calling the machine language routine from BASIC.

a) Use of USR (10K, BASIC)

1. Write 4C 00 03 from the address of 000A (Hexadecimal). This is a jump instruction to the address of 300 (Hexadecimal).
2. Write the following machine language routine starting with address 300 (Hexadecimal):

STRT :	JSR	$E10C	20	0C	E1
	LDA	$A1	A5	A1	
LOOP :	BIT	$C1C1	2C	C1	C1
	BMI	LOOP	30	FD	
	STA	$C090	8D	90	C0
	RTS		60		

3. For example, if you want to transfer a code "75" three times;
   10 DUMMY = USR (75) + USR (75) + USR (75).
   (DUMMY is an arbitrary variable name)

[EXAMPLE]

```
]LIST

100  REM  BIT-IMAGE 2
110 :
120  REM  DATA TRANSFER SUBROUTIN
     E
130  RESTORE :AD = 768
140  FOR I = 1 TO 14
150  READ A: POKE AD,A:AD = AD +
     1
160  NEXT
170  DATA  32,12,225,165,161,44,1
     93,193,48,251,141,144,192,96

180 :
190  REM  JMP VECTOR
200  POKE 10,76: POKE 11,0: POKE
     12,3
210 :
220  REM  ESC+K+(8)+(0)
230 Z =  USR (27) +  USR (75) +  USR
     (8) +  USR (0)
240 :
250  REM  BIT-IMAGE DATA
260 Z =  USR (34) +  USR (82) +  USR
     (138) +  USR (143) +  USR (1
     38) +  USR (82) +  USR (34) +
      USR (0)
270 :
280 Z =  USR (10) +  USR (10): REM
      LF CODE




]RUN
```

LIST 100 — 200: Program to write the aforementioned machine language from 300 (Hex.) address.

LIST 210 — 280: Program to output the Bit Image date to the printer.

b) Use of CALL

1. Write the following machine language routine from address 300:

STRT :	LDA	$00	A5	00	
LOOP :	BIT	$C1C1	2C	C1	C1
	BMI	LOOP	30	FB	
	STA	$C090	8D	90	C0
	RTS		60		

Note: $00 is an address to store data to be output monentarily onto the printer.

[Example]

```
]LIST

100  REM  BIT-IMAGE 3
110 :
120  REM   DATA TRANSFER SUBROUTI
     NE
130  RESTORE :AD = 768
140  FOR I = 1 TO 11
150  READ A: POKE AD,A:AD = AD +
     1
160  NEXT
170  DATA  165,0,44,193,193,48,25
     1,141,144,192,96
180 :
190  REM  ESC+K+(8)+(0)
200  POKE 0,27: CALL 768
210  POKE 0,75: CALL 768
220  POKE 0,8: CALL 768
230  POKE 0,0: CALL 768
240  POKE 0,34: CALL 768
250  POKE 0,82: CALL 768
260  POKE 0,138: CALL 768
270  POKE 0,143: CALL 768
280  POKE 0,138: CALL 768
290  POKE 0,82: CALL 768
300  POKE 0,34: CALL 768
310  POKE 0,0: CALL 768
320 :
330  POKE 0,10: CALL 768
340  POKE 0,10: CALL 768



]RUN
```

LIST 100 — 170: Program to input machine language routine.
LIST 190 — 340: Program to output the Bit Image data to the printer.

2. If you want to transfer, for example, a code "75" three times;

```
10    POKE    0,75:    CALL    768
20    POKE    0,75:    CALL    768
30    POKE    0,75:    CALL    768
```

For further details of printer functions, see the respective Operation Manuals.

3. Bit Image Applications

This section deals with some program samples utilized by Bit Image.

a) Emphasized printing

The emphasized printing is achieved by printing the same text twice, which second print is shifted by some dots horizontally from the first print.

[EXAMPLE]

```
]LIST

100  REM  EMPHASIZED PRINTING
110 :
120  PR# 1
130 SLOT = 1
140  POKE 1656 + SLOT,72: REM  OU
     TPUT ONLY
150  POKE 1528 + SLOT,255: REM  K
     ILL LF
160  PRINT "EMPHASIZED PRINTING"
170  PRINT  CHR$ (27); CHR$ (75);
      CHR$ (1); CHR$ (0); CHR$ (0
     );
180  PRINT "EMPHASIZED PRINTING"
190  POKE 1528 + SLOT,0: REM  SEN
     D LF CODE
200  POKE 1657 + SLOT,40: REM  OU
     TPUT CRT
210  PR# 0

]RUN


EMPHASIZED PRINTING
```

b) Underlining
Any underlining may be achieved as in the following example by combining Bit Image mode printing and Text mode printing.

[EXAMPLE]

```
]LIST

100  REM  UNDER-LINED PRINTING
110 :
120  PR# 1
130 SLOT = 1
140  POKE 1656 + SLOT,72
150  POKE 1528 + SLOT,255
160  PRINT "UNDER-LINED PRINTING"

170  PRINT  CHR$ (27); CHR$ (75);
      CHR$ (11 * 6); CHR$ (0);
180  FOR I = 1 TO 11 * 6: PRINT  CHR$
     (1);: NEXT
190  PRINT  CHR$ (10)
200  POKE 1528 + SLOT,0
210  POKE 1657 + SLOT,40
220  PR# 0



]RUN


UNDER-LINED PRINTING
```

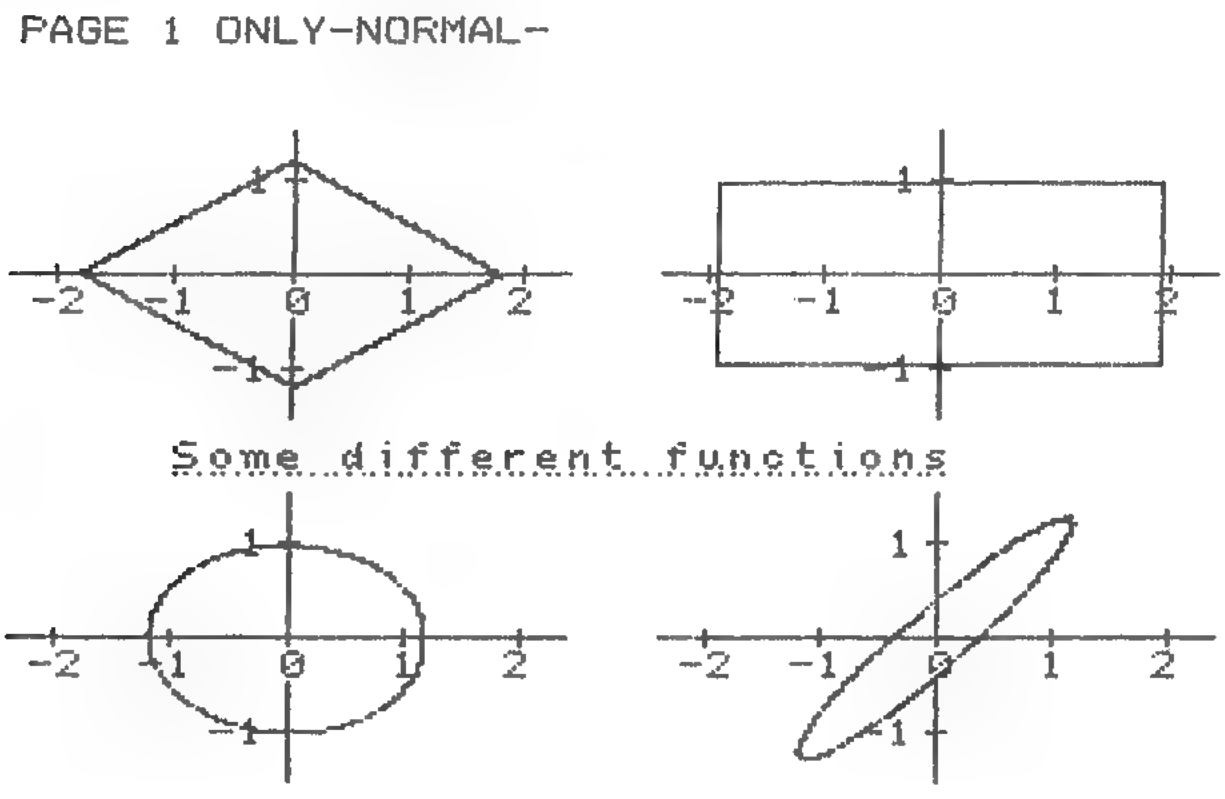
HARD COPY OF HI-RES GRAPHICS

This chapter explains how to get the hard copy of Hi-resolution graphics of APPLE II is described.
To obtain the hard copy, the general practice is to set a certain value in the memory called ■ mode register. APPLE II has two memories for HI-RES graphics, called page 1 and page 2. Logical operations such as AND, OR, and EOR between these two memories can be achieved by setting a proper value in the mode register. The following explains the mode register setting with printing samples. After setting the mode register, printing of the HI-RES image is started by presetting "CTRL-Q". "CTRL-Q" is thus an instruction for making hard copy of the HI-RES image.

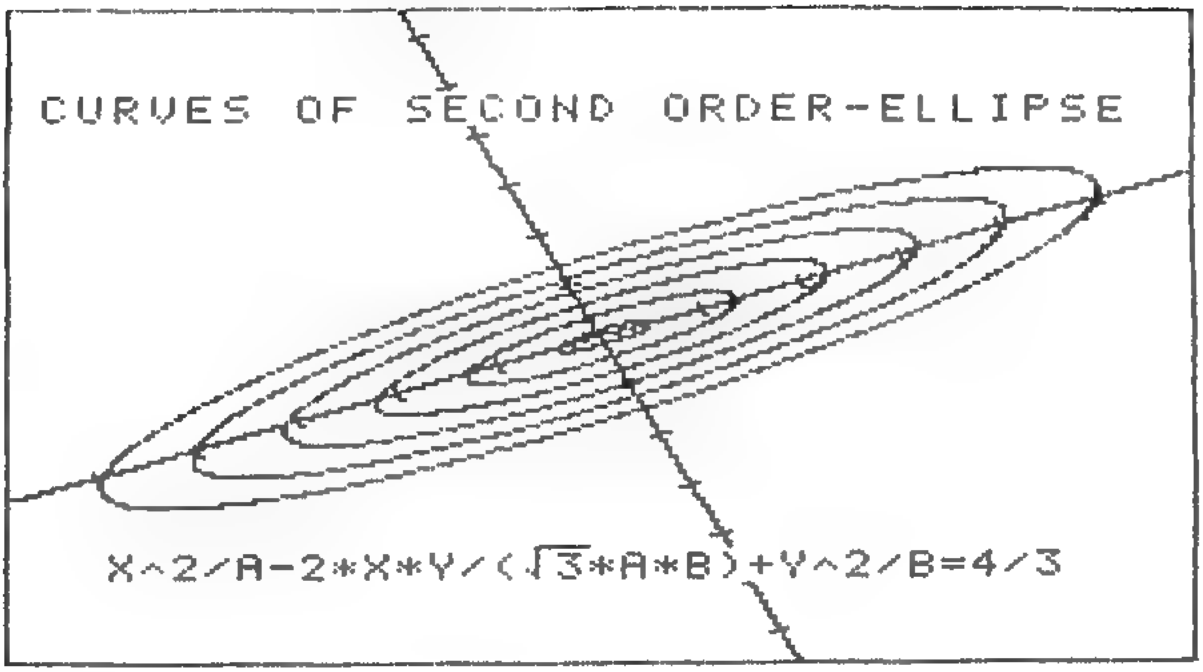
1. Data Stored in the Mode Register

1.1. Selection of HI-RES page [P]

This parameter specifies the page to be copied.
Page 1 : P = 1
Page 2 : P = 2
Both pages : P=3 (Horizontal parallel printing)



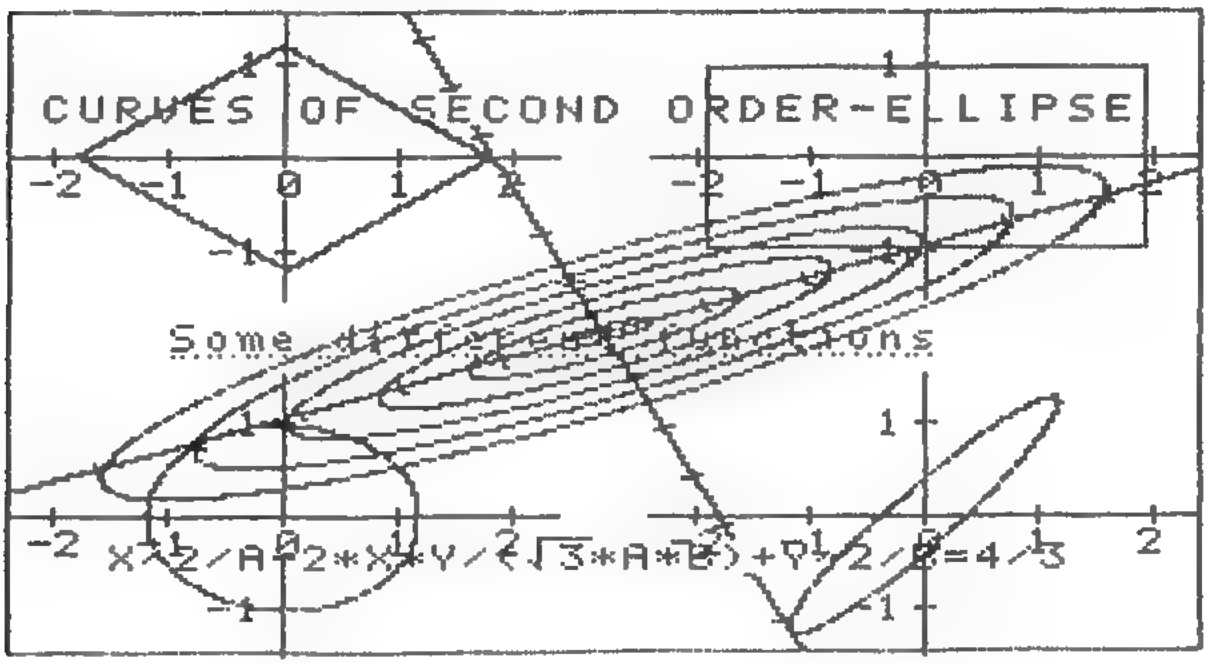
PAGE 2 ONLY-NORMAL-



1.2. Logical operation [L]

Logical product (AND) : L = 4
Logical sum (OR) : L = 8
Exclusive-or operation (EOR) : L = 16
Logical operations are executed bit by bit between the corresponding data of page 1 and page 2.
When L = 0, no operation is executed.

PAGE 1 'OR' PAGE 2



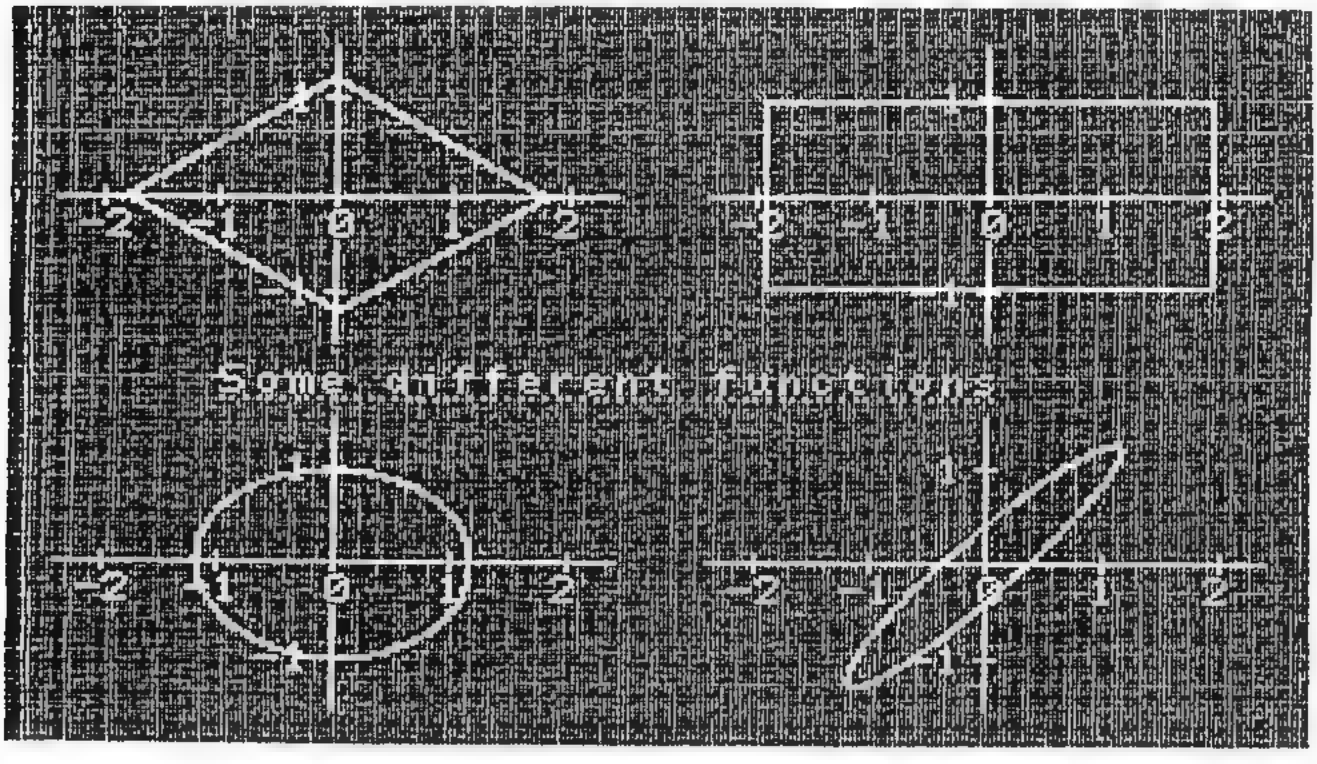
1.3. Inverse printing instruction [I]

An image on a specified page is printed with the color inversed (black-white).

Normal: I = 0

Inverse: I = 32

PAGE 1 (INVERSE)



1.4. Printing by line units [U]

A number of lines on the HI-RES screen corresponding to that on the text mode is printed. This is specified by TAB and VTAB in line units consisting of 8 vertical dots and 280 horizontal dots. Printing is done for 40 blocks × 24 lines in units of 7 × 8 dots.

Image unit : U = 0

Line unit : U = 128

```
  WHOLE CHRACTER PICTURE

 1    THIS IS
 3     THIS IS A
 5      THIS IS AN
 7       THIS IS AN
 9        THIS IS AN E
11         THIS IS AN EX
13          THIS IS AN EXA
15           THIS IS AN EXAM
17            THIS IS AN EXAMP
19             THIS IS AN EXAMPL
21              THIS IS AN EXAMPLE
23               THIS IS AN EXAMPLE.

  LINE 19-23'S PRINTING


19             THIS IS AN EXAMPL

21              THIS IS AN EXAMPLE

23               THIS IS AN EXAMPLE.
```

LINE 19-23'S PRINTING

```
19          THIS IS AN EXAMPL

21           THIS IS AN EXAMPLE

23            THIS IS AN EXAMPLE.
```

19 THIS IS AN EXAMPL

21 THIS IS AN EXAMPLE

23 THIS IS AN EXAMPLE.

ESC 2 and CR codes are automatically sent after printing in this interface. Specify the line spacing as follows.

```
10 Q$ = CHR$ (17) :  REM CTRL-Q
20 PRINT Q$; CHR$ (27) ; "A"; CHR$ (J)
```

Where "J" represents the amount of line feed.

1.5. Enlarged printing [D]

A HI-RES screen is printed by enlarging twice both horizontally and vertically.
Enlarging : D = 64
Standard : D = 0

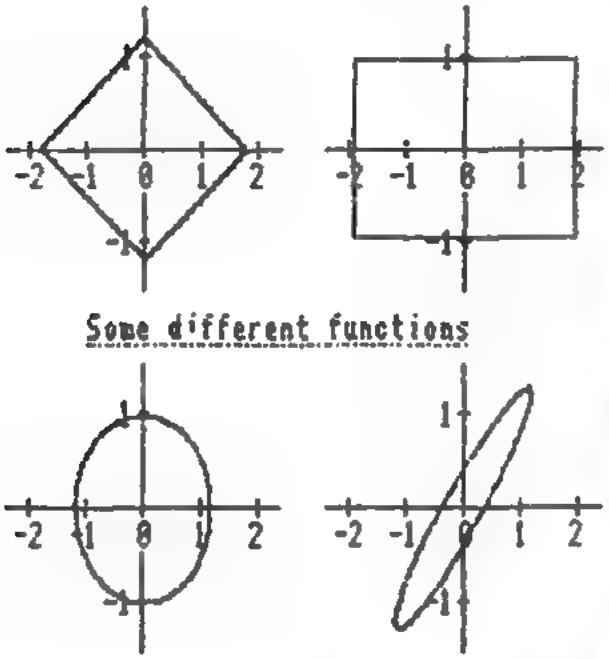
1.6. Dual density Bit Image printing [DD]

EPSON Dot Matrix Printer has a dual density Bit Image mode which makes finer graphics on the paper than the normal density mode.
Instruction comes through an address other than the mode register.
Dual Density: DD = 76 POKE 114 + n, DD
Normal Density: DD = 75

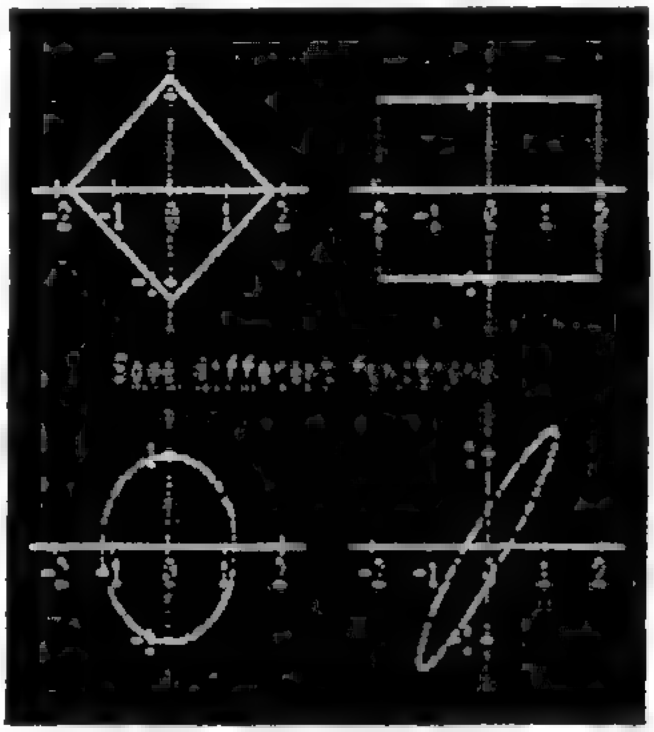
[Example]

```
10 POKE 1144 + n, 76
20 PRINT CHR$(17): REM CTRL-Q
30 POKE 1144 + n, 75
```

PAGE 1 ONLY-NORMAL--



PAGE 1 (INVERSE)



2. Mode Register Setting

Mode register refers to the 8-bit data present in the address of 1912 + n as shown below;

MSB LSB

Line Unit	Enlarge	Inverse	EOR	OR	AND	HI-RES Page 2	HI-RES Page 1
128	64	32	16	8	4	2	1

Since the mode register has been assigned as described above, the required mode is obtained by writing the sum of the values described in the above section 1.1. through 1.6.

POKE 1912 + n, P+L+I+U+D

[Example]

```
10 POKE 1912 + n, 98 (Print the page 2 by Enlarging and Inversing)
20 PRINT CHR$ (17)
30 VTAB 15
40 POKE 1212 + n, 129 (Print the 15th line of page 1)
50 PRINT CHR$ (17)
```

Note: The value of the mode register is 1 when the slot in which this interface is inserted is specified. If an unfeasible or contradictory mode is given, it may interfere with the execution of the program.

[Remark]

Firmware area

A 2-k byte ROM is used for C800 — CFFF and CNXX. A correct access to CFFF is important to disable this ROM. It should be noted that a simple operation of OP^{c} (Zero, Control P) or PR#0 does not disable the firmware. The formatting subroutine may therefore be used with the printer "off".

PRINT FORMATTING ROUTINE

1. Outline

When a variable is displayed in the APPLE'S software, a format may be specified as shown following example; the formatting routine is added to the Applesoft BASIC. It is helpful to print out data in a predetermined format.

1) Integer: In (Example: I3 → 123)
2) Fixed point representation: Fn.m (F4.2 → 42.51)
3) Exponent: En (E3 → −5.43 E-01)

n: Includes only the number of numeric characters and does not include the number of codes, decimal point and exponent. In the range of $1 \leq n \leq 12$, a space comes first if the actual number of digits is smaller than n.

m: Includes only the number of numeric characters below the decimal point.
It must be $0 \leq m < n$.

2. How to Use in a Program

1. A ROM 2716 (2-k byte) is incorporated in the interface board and its address is assigned for $C800-CFFF. This ROM is started by PR#n. Where n is slot number.
   An instruction to stop printing, PR#0, will not disable the ROM. For this reason the print formatting subroutine may be used in the user's program even though the printer is not in use.
   To disable the ROM on the board, execute; POKE-12289, 0 or STA $CFFF.
2. Prior to program execution, run the following statement;
   PR#SLOT
   PR#0 (to stop printing)
   WRITE = 49312 + 256 *SLOT
3. Specification of the print format
   A. Calling a subroutine ——(apply it when line feed is required)
      CALL WRITE: CHR$(13):
      Enclose the variable name, format etc, to be printed by a beginning colon (:) and a closing colon (:).
   B. Integer, fixed point representation and exponent
      1) Integer ..... CALL WRITE : X% ; I4 :
      2) Fixed point representation ..... CALL WRITE : X ; F5.2 :
      3) Exponent ..... CALL WRITE : X ; E6 :

      Insert a semicolon (;) between the variable and the format.

C. Break point symbol and spacing

Since a break point (,)/ does not provide a sufficient space in printing, use a double quotation mark (").

```
CALL WRITE : I ; I3, "   ", Y (I) ; F8.2, "   ", Z(I) ; E4:
```

Use a comma (,) as a break point symbol.

D. String

If string length is constant.

```
CALL WRITE : J% ; I2, "      SIN=", SX ; F6.5 :
CALL WRITE : K ;I2, "   ", S$, SX ; F6.5 :
```

If string length is variable.

Note: The number of letters allocated for the string variable is 0 to 255.

Do not use the wrong format.

E. Array format

```
CALL WRITE : A% (I) ; I3, "   ", B (I, J, K,) ; F12.8 :
```

These can be handled in the same way as a normal variable.

F. Miscellaneous

1) Line feed is not executed by statement "CALL WRITE : : "
2) A variable which has not been specified for formatting is printed in the same format as that of the preceding variable.

   ```
   CALL WRITE : I ; I3, "   ", X :
   ```

   X is printed in the same format as that of I3.
3) If the specified format is incorrect, it will be neglected during printing.

   Incorrect : I13, F6, E20, etc.
4) If a value overflows the specified format, *** ... * is printed.
5) If you want the line feed after execution, send CHR$ (13).

```
]LIST

100  REM  PRINT FORMATING DEMO
110 :
120 WRITE = 52480
140 :
150  REM  VIDEO ONLY
160  REM  INIT FORMATTING PROG.
170  PR# 1: PRINT  CHR$ (0);
175  PR# 0
180  GOSUB 1000
190  REM  PRINTER ONLY
200  PR# 1: PRINT  CHR$ (0);
210  GOSUB 1000
220  REM  FORMATTING DEACTIVE
230  POKE 53247,255
240  END
250 :
260 :
1000  REM  FORMATTING
1010  PRINT "   ***  PRINT FORMAT
     TING DEMO ***": PRINT
1015  PRINT "   X           Y=EXP(
     X)"
1020  PRINT " -I3-    -F8.3-    -
     I7-        -E6-": PRINT
1025  FOR I =  - 7 TO 12
1030 Y =  EXP (I)
1040  CALL WRITE:I;I3," ":
1050  CALL WRITE:Y;F8.3," ":
1060  CALL WRITE:Y;I6,"  ":
1070  CALL WRITE:Y;E6, CHR$ (13):

1080  NEXT
1090  RETURN

]
```

*** PRINT FORMATTING DEMO ***

X	Y=EXP(X)		
-I3-	-F8.3-	-I7-	-E6-
-7	0.000	0	9.11881E-04
-6	0.002	0	2.47875E-03
-5	0.006	0	6.73794E-03
-4	0.018	0	1.83156E-02
-3	0.049	0	4.97870E-02
-2	0.135	0	1.35335E-01
-1	0.367	0	3.67879E-01
0	1.000	1	1.00000E+00
1	2.718	2	2.71828E+00
2	7.389	7	7.38905E+00
3	20.085	20	2.00855E+01
4	54.598	54	5.45981E+01
5	148.413	148	1.48413E+02
6	403.428	403	4.03428E+02
7	1096.633	1096	1.09663E+03
8	2980.957	2980	2.98095E+03
9	8103.083	8103	8.10308E+03
10	22026.465	22026	2.20264E+04
11	59874.141	59874	5.98741E+04
12	**********	162754	1.62754E+05

Z-80 SOFT CARD (Microsoft Inc.)

Together with a Z-80 Soft Card, you can operate the printer (LPT:) by giving the following instructions (Install the interface board in slot 1):

```
A > DDT
DDT VER 2.2
-SDD2F*
DD2F 3E 31
DD30 DD
- (control C)
A >
```

*Note: In CP/M 44K version, input SAD 2F instead of SDD 2F.

APPLE WRITER

"Apple Writer", a program by APPLE COMPUTER INC., can specify the printing format to meet the requirements of various printers.
Make sure to specify MSB at 1 when using it.
In the original "Apple Writer", default value is specified at 1.

APPLE PLOT

Modify the greeting program "HELLO" by the procedure described below to use an "APPLE PLOT".

Step
1. Boot the system with another DOS 3.2 diskette.
2. Insert the diskette of APPLE PLOT into the disk drive.
3. Load HELLO from the APPLE PLOT Master diskette.
4. Make three modifications in the line No. 10 as shown in the next EXAMPLE.
5. Type

```
UNLOCK HELLO
SAVE HELLO
LOCK HELLO
```

Operation

Step
1. Choose No. 9 in the "MENU".
2. Choose "1. APPLE SILENTYPE".
3. Answer "N" at OPTION.
4. Press {CTRL} C keys to stop printing.

[EXAMPLE]

```
]LIST

0  PRINT  CHR$ (4);"PR#0"
1  PRINT  CHR$ (4);"IN#0"
2  TEXT : HOME
4  VTAB (10): PRINT  TAB( 12): INVERSE
     : PRINT "LOADING APPLE PLOT"
     : NORMAL
6  PRINT "NOMON C,I,O": PRINT "MA
     XFILES 1"
8  FOR J = 672 TO 729: READ K: POKE
     J,K: NEXT J: GOTO 12
10  DATA    169,0,133,30,169,192,
     133,31,230,31,165,31,201,200
     ,176,37,160,4,177,30,201,138
      ,208,240,160,19,177,30,201,
     232,208,232,160,35,177,30,20
     1, 41,208,224,165,31,41,15,1
     33,0,32,149,254,32,234,3,96,
     169,0,133,0,96
12  DATA   172,192,3,185,13,200,1
     41,192,3,96:. FOR I = 961 TO
     970: READ K: POKE I,K: NEXT

14  PRINT  CHR$ (4);"RUN APPLEPLO
     T"
```

CIRCUIT DIAGRAM OF INTERFACE BOARD

Fig. 5 shows the circuit diagram of the interface board.

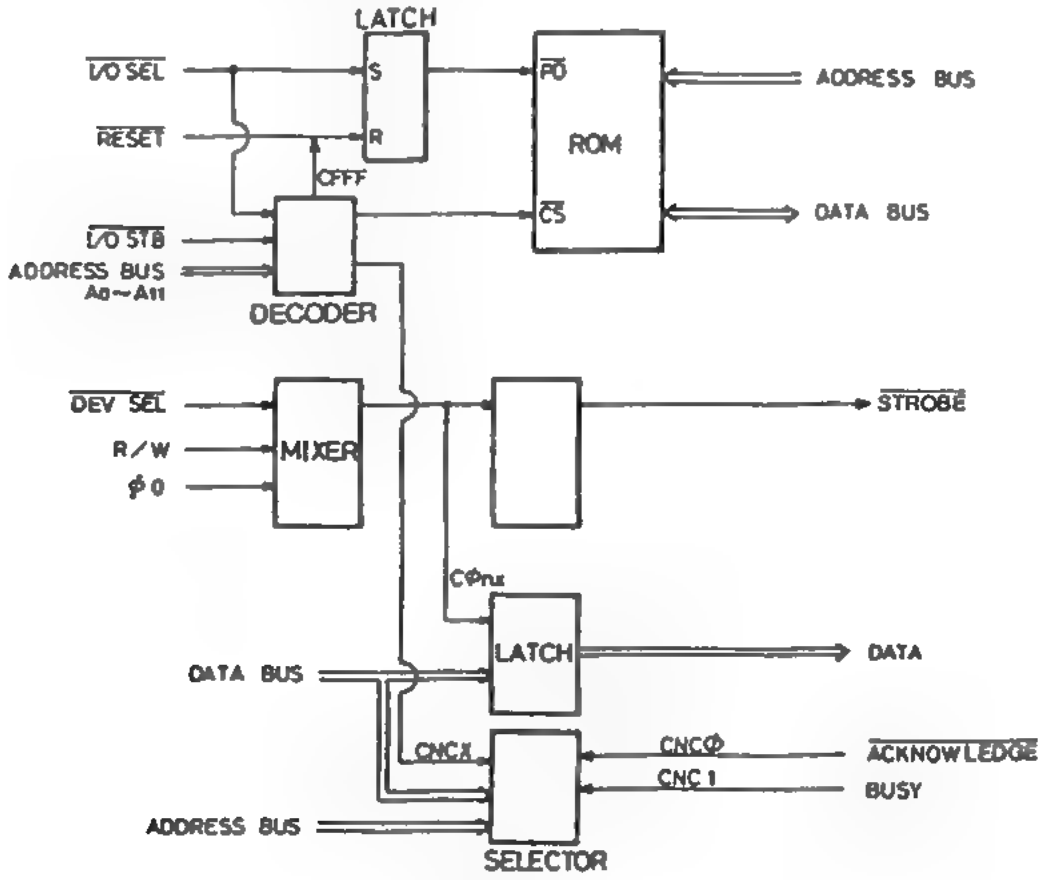


Fig. 5 CIRCUIT DIAGRAM OF INTERFACE BOARD

There is a ROM with 2-k bytes on the interface board which contains programs to access data between your EPSON Printer and APPLE II. Data output to the printer is mode by executing a store command to C0nx address (where n is 8 + slot, and X is an arbitrary value). For example, if the interface board is installed in slot 1, then the data is transferred to the printer by executing.

```
LDA    DATA
STA    CØ9Ø.
```

$\overline{\text{ACKNOWLEDGE}}$ and BUSY signals refer the Most Significant Bits in the CNC0 and CNC1 address respectively. N represents the slot number.

C800 to CFFF are absolute addresses which may be used freely in APPLE II. Programs in this area tend to be enabled by accerssing CNXX and to be disabled by accessing the specific address, CFFF. CFFF should be accessed after use of the routine in C800 to CFFF.

NOTICE

In this interface Mode Register (1272 + SLOT address No.) is initialized by executing PR#1 for the first time after the POWER is turned ON. Therefore the content of Mode Resister will not be initialized again once PR# should be executed.

There might be cases when Mode Register is not initialized, and subsequently cause the erroneous operation. In this case please execute first,

```
POKE 1272 + SLOT, 255
```

Then execute

```
PR#1
```

When the printing of the lower cases are interrupted by RESET please execute the following to clear the lower case flag.

```
POKE 1784 + SLOT, 0
```

Some functions including the copies of High Resolution screen cannot be performed in APPLE PASCAL, and CP/M.

CONNECTOR PIN ASSIGNMENT TABLES

1. I/F BOARD SIDE

PIN NO.	SIGNAL	WIRE COLOR
1	STROBE	RED
2	GND	GRAY
3	DATA 1	GRAY
4	GND	GRAY
5	DATA 2	YELLOW/GREEN
6	GND	GRAY
7	DATA 3	GRAY
8	GND	GRAY
9	DATA 4	GRAY
10	GND	YELLOW/GREEN
11	DATA 5	GRAY
12	GND	GRAY
13	DATA 6	GRAY
14	GND	GRAY
15	DATA 7	YELLOW/GREEN

PIN NO.	SIGNAL	WIRE COLOR
16	GND	GRAY
17	DATA 8	GRAY
18	GND	GRAY
19	$\overline{\text{ACK}}$	GRAY
20	BUSY	YELLOW/GREEN

2. PRINTER SIDE

PIN NO.	SIGNAL
1	STROBE
2	DATA 1
3	DATA 2
4	DATA 3
5	DATA 4
6	DATA 5
7	DATA 6
8	DATA 7
9	DATA 8
10	ACK
11	BUSY
12 — 18	N/A
19 — 27	GND
28 — 36	N/A

APPENDIX

	Decimal	Hexadecimal
Printing Column	1656 + n	$678 + n
Codes which specify the lower cases	1400 + n	$578 + n
Specify output of LF code	1528 + n	$5F8 + n

Note: n represents a slot number.

Schematic Diagram

Fig. 6 shows the schematic diagram of the Interface Board.

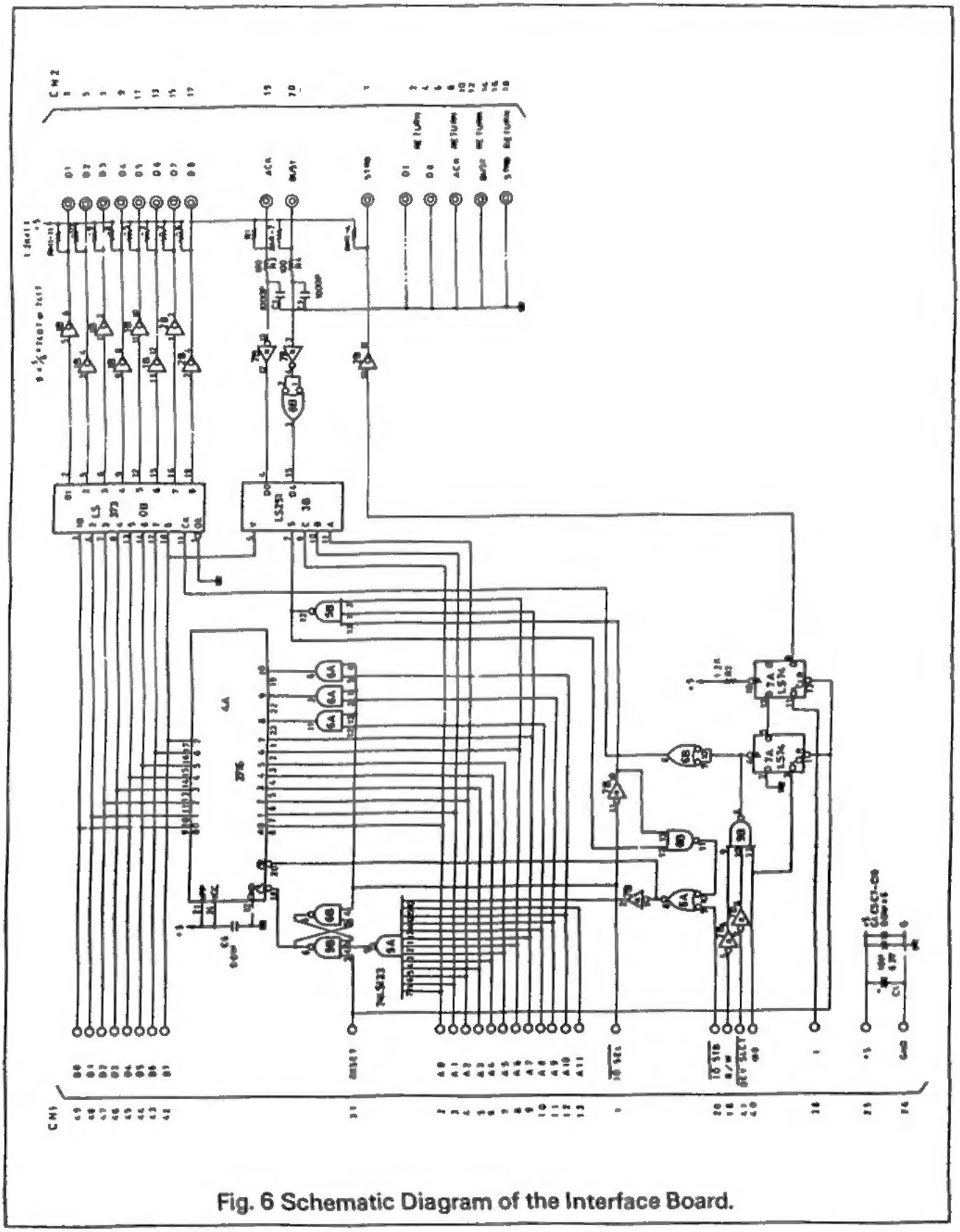


Fig. 6 Schematic Diagram of the Interface Board.

LCC 8132 EPSON EPROM

LEICESTER COMPUTER CENTRE
67 REGENT ROAD
LEICESTER LE1 6YF

TEL 0533 556268